# Advanced Statistics Project Report

# Contents

# List of Figures



# List of Equations



# List of Tables

# Problem 1

Salary is hypothesized to depend on educational qualification and occupation. To understand the dependency, the salaries of 40 individuals [SalaryData.csv ] are collected and each person's educational qualification and occupation are noted. Educational qualification is at three levels, High school graduate, Bachelor, and Doctorate. Occupation is at four levels, Administrative and clerical, Sales, Professional or specialty, and Executive or managerial. A different number of observations are in each level of education – occupation combination.

[Assume that the data follows a normal distribution. In reality, the normality assumption may not always hold if the sample size is small.]

## Problem 1A

### 1.1 State the null and the alternate hypothesis for conducting one-way ANOVA for both Education and Occupation individually.

Null hypothesis (H0) is the presumed current state of the matter or status quo. Unless there is strong evidence otherwise, null hypothesis is not rejected. Alternative hypothesis (HA) is the rival opinion or research hypothesis

***a) Null hypothesis related to Education***

H0: There is no effect of education on salary

HA: For at least one education level, mean salary is different from the others

$H0: \mu H = \mu B = \mu D$

$HA$: At least one mean is different from the others

***b) Null hypothesis related to Occupation***

H0: There is no effect of occupation on salary

HA: For at least one occupation level, mean salary is different from the others

$H0: \mu A = \mu S = \mu P = \mu E$

$HA$: At least one mean is different from the others

1.2 Perform a one-way ANOVA on Salary with respect to Education. State whether the null hypothesis is accepted or rejected based on the ANOVA results.

**First deciding the level of significance:**

The level of significance is defined as the probability of rejecting a null hypothesis when it is true, and is denoted by α. That is, P (Type I error) = α = 0.05

The below is output from F_oneway from scipy.stats.

```
F_onewayResult(statistic=30.95, pvalue=1.257e-08)
```

**Interpretation**: The P-value obtained from ANOVA for Education is statistically significant as (P < 0.05). We conclude Salary depends on the levels of Education. Mean salary for at least one level of Education is different from the others.

1.3 Perform a one-way ANOVA on Salary with respect to Occupation. State whether the null hypothesis is accepted or rejected based on the ANOVA results.

The below is output from F_oneway from scipy.stats.

```
F_onewayResult(statistic=0.88, pvalue=0.458)
```

**Interpretation:** The null hypothesis that mean salary is different for different level of occupation is not rejected.

1.4 If null hypothesis is rejected in either (1.2) or in (1.3), find out which class means are significantly different. Interpret the result.

The null hypothesis that mean salary is different for different levels of educational qualification is rejected, but the null hypothesis that mean salary is different for different level of occupation is not rejected. Hence we need to find out for which educational qualification mean salary is significantly different from the other levels.

Pairwise Tukey HSD test is performed as below-

Multiple Comparison of Means - Tukey HSD, FWER=0.05

| group1 | group2 | meandiff | p-adj | lower | upper | reject |
|---|---|---|---|---|---|---|
| Bachelors | Doctorate | 43274.06 | 0.014 | 7541.14 | 79006.98 | True |
| Bachelors | HS-grad | -90114.15 | 0.001 | -132035.19 | -48193.11 | True |
| Doctorate | HS-grad | -133388.22 | 0.001 | -174815.08 | -91961.35 | True |

From the output it is clear that mean salary is significantly different for each pair of means.

## Problem 1B

1.5 What is an interaction between two treatments? Analyse the effects of one variable on the other (Education and Occupation) with the help of an interaction plot. [hint: use the 'interaction_plot' function from the 'statsmodels.graphics.factorplots' module]

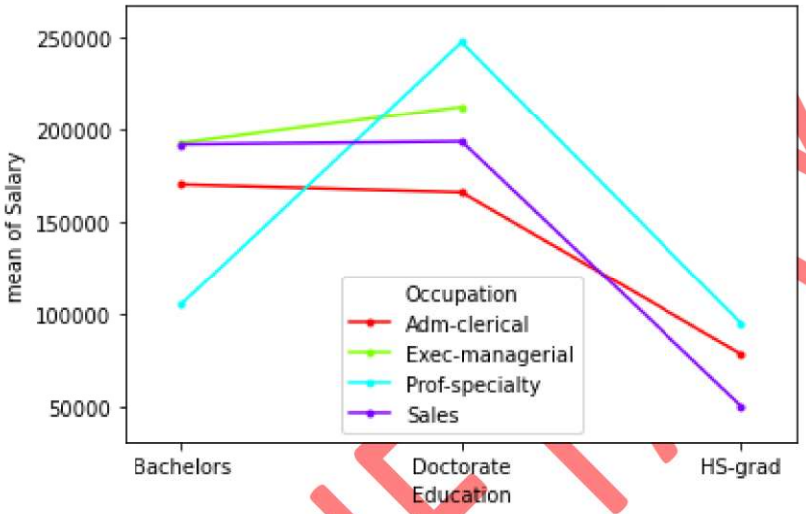



*Figure 1- Interaction Plot*

Interaction exists when the effect of one variable depends on the value of the second variable. Interaction modifies the relationship between the dependent and the independent variable, according to the value of a third variable. This type of effect makes the model more complex, but if the real world behaves this way, it is critical to incorporate it in your model. In this interaction plot, the lines are not parallel. Thus interaction between Education and Occupation is indicated.

1.6 Perform a two-way ANOVA based on Salary with respect to both Education and Occupation (along with their interaction Education*Occupation). State the null and alternative hypotheses and state your results. How will you interpret this result?

**In case of two-way ANOVA both sets of hypotheses are considered simultaneously.**

$H0$: $\mu H = \mu B = \mu D$

$HA$: At least one mean is different from the others

$H0$: $\mu A = \mu S = \mu P = \mu E$

$HA$: At least one mean is different from the others

*Level of Significance* $\alpha = 0.05$

| | df | sum_sq | mean_sq | F | PR(>F) |
|---|---|---|---|---|---|
| **Education** | 2.0 | 1.02e+11 | 5.13e+10 | 72.21 | 5.46e-12 |
| **Occupation** | 3.0 | 5.51e+09 | 1.83e+09 | 2.587 | 7.21e-02 |
| **Education:Occupation** | 6.0 | 3.63e+10 | 6.058e+09 | 8.51 | 2.23e-05 |
| **Residual** | 29.0 | 2.06e+10 | 7.11e+08 | NaN | NaN |

***NOTE: The p-value of interaction here is different from the result in R as the degrees of freedom for interaction here is considered as 6 as opposed to 5 in R***

**Interpretation**: The essential results do not change from one-way ANOVA. Education is still a significant differentiator for Salary, but Occupation is not. However, note that the P-values change. The interaction effects are also significant

### 1.7 Explain the business implications of performing ANOVA for this particular case study.

ANOVA test compares group means of more than two groups at the same time to determine whether all group means are equal or not. This is another way to determine whether the discrete predictor does indeed have an influence on the continuous response.

- This particular problem deals with the dependence of Salary on Education and Occupation. Education is at 3 levels and Occupation is at 4 levels.
- ANOVA result indicates that Salary depends on Education but not on Occupation.
- Education and Occupation have an interaction effect. That means some levels of Occupation is more suitable for persons with certain levels of Education.
- The interaction effect is also significant.

# Problem 2

The dataset Education - Post 12th Standard.csv contains information on various colleges. You are expected to do a Principal Component Analysis for this case study according to the instructions given. The data dictionary of the 'Education - Post 12th Standard.csv' can be found in the following file: Data Dictionary.xlsx.

## Perform Exploratory Data Analysis [both univariate and multivariate analysis to be performed]. What insight do you draw from the EDA?

Dataset Head:

| | 0 | 1 | 2 | 3 | 4 |
|---|---|---|---|---|---|
| **Names** | Abilene Christian University | Adelphi University | Adrian College | Agnes Scott College | Alaska Pacific University |
| **Apps** | 1660 | 2186 | 1428 | 417 | 193 |
| **Accept** | 1232 | 1924 | 1097 | 349 | 146 |
| **Enroll** | 721 | 512 | 336 | 137 | 55 |
| **Top10perc** | 23 | 16 | 22 | 60 | 16 |
| **Top25perc** | 52 | 29 | 50 | 89 | 44 |
| **F.Undergrad** | 2885 | 2683 | 1036 | 510 | 249 |
| **P.Undergrad** | 537 | 1227 | 99 | 63 | 869 |
| **Outstate** | 7440 | 12280 | 11250 | 12960 | 7560 |

| | 0 | 1 | 2 | 3 | 4 |
|---|---|---|---|---|---|
| **Room.Board** | 3300 | 6450 | 3750 | 5450 | 4120 |
| **Books** | 450 | 750 | 400 | 450 | 800 |
| **Personal** | 2200 | 1500 | 1165 | 875 | 1500 |
| **PhD** | 70 | 29 | 53 | 92 | 76 |
| **Terminal** | 78 | 30 | 66 | 97 | 72 |
| **S.F.Ratio** | 18.10 | 12.20 | 12.90 | 7.70 | 11.90 |
| **perc.alumni** | 12 | 16 | 30 | 37 | 2 |
| **Expend** | 7041 | 10527 | 8735 | 19016 | 10922 |
| **Grad.Rate** | 60 | 56 | 54 | 59 | 15 |

There are total 777 rows and 18 columns in the dataset.

| # | Column | Non-Null | Count | Dtype |
|---|---|---|---|---|
| 0 | Apps | 777 | non-null | float64 |
| 1 | Accept | 777 | non-null | float64 |
| 2 | Enroll | 777 | non-null | float64 |
| 3 | Top10perc | 777 | non-null | float64 |
| 4 | Top25perc | 777 | non-null | float64 |
| 5 | F.Undergrad | 777 | non-null | float64 |
| 6 | P.Undergrad | 777 | non-null | float64 |
| 7 | Outstate | 777 | non-null | float64 |
| 8 | Room.Board | 777 | non-null | float64 |
| 9 | Books | 777 | non-null | float64 |
| 10 | Personal | 777 | non-null | float64 |
| 11 | PhD | 777 | non-null | float64 |
| 12 | Terminal | 777 | non-null | float64 |

| | | | | |
|---|---|---|---|---|
| 13 | S.F.Ratio | 777 | non-null | float64 |
| 14 | perc.alumni | 777 | non-null | float64 |
| 15 | Expend | 777 | non-null | float64 |
| 16 | Grad.Rate | 777 | non-null | float64 |

From the info of the data, we can infer that there are no null values in the dataset. All the variables in the dataset are continuous except Names.

| | count | mean | std | min | 25% | 50% | 75% | max |
|---|---|---|---|---|---|---|---|---|
| **Apps** | 777.00 | 3001.64 | 3870.20 | 81.00 | 776.00 | 1558.00 | 3624.00 | 48094.00 |
| **Accept** | 777.00 | 2018.80 | 2451.11 | 72.00 | 604.00 | 1110.00 | 2424.00 | 26330.00 |
| **Enroll** | 777.00 | 779.97 | 929.18 | 35.00 | 242.00 | 434.00 | 902.00 | 6392.00 |
| **Top10perc** | 777.00 | 27.56 | 17.64 | 1.00 | 15.00 | 23.00 | 35.00 | 96.00 |
| **Top25perc** | 777.00 | 55.80 | 19.80 | 9.00 | 41.00 | 54.00 | 69.00 | 100.00 |
| **F.Undergrad** | 777.00 | 3699.91 | 4850.42 | 139.00 | 992.00 | 1707.00 | 4005.00 | 31643.00 |
| **P.Undergrad** | 777.00 | 855.30 | 1522.43 | 1.00 | 95.00 | 353.00 | 967.00 | 21836.00 |
| **Outstate** | 777.00 | 10440.67 | 4023.02 | 2340.00 | 7320.00 | 9990.00 | 12925.00 | 21700.00 |
| **Room.Board** | 777.00 | 4357.53 | 1096.70 | 1780.00 | 3597.00 | 4200.00 | 5050.00 | 8124.00 |
| **Books** | 777.00 | 549.38 | 165.11 | 96.00 | 470.00 | 500.00 | 600.00 | 2340.00 |
| **Personal** | 777.00 | 1340.64 | 677.07 | 250.00 | 850.00 | 1200.00 | 1700.00 | 6800.00 |
| **PhD** | 777.00 | 72.66 | 16.33 | 8.00 | 62.00 | 75.00 | 85.00 | 103.00 |

| | count | mean | std | min | 25% | 50% | 75% | max |
|---|---|---|---|---|---|---|---|---|
| **Terminal** | 777.00 | 79.70 | 14.72 | 24.00 | 71.00 | 82.00 | 92.00 | 100.00 |
| **S.F.Ratio** | 777.00 | 14.09 | 3.96 | 2.50 | 11.50 | 13.60 | 16.50 | 39.80 |
| **perc.alumni** | 777.00 | 22.74 | 12.39 | 0.00 | 13.00 | 21.00 | 31.00 | 64.00 |
| **Expend** | 777.00 | 9660.17 | 5221.77 | 3186.00 | 6751.00 | 8377.00 | 10830.00 | 56233.00 |
| **Grad.Rate** | 777.00 | 65.46 | 17.18 | 10.00 | 53.00 | 65.00 | 78.00 | 118.00 |

Looking at the mean values from the 5-point summary, we can see that on an average approx. 3000 applications are received in US universities, out of which around 2020 applications are accepted by the universities and around 780 new students get enrolled.

The average cost for room and board is approx. 4350, for books its around 550 and for personal expense its around 1350.

Moreover, the average number of full time undergrad students are around 3700 whereas the average number of part-time undergrad students stand low at around 850.

## Univariate Analysis

*Note: Median – White and Mean – Yellow in below plots*

Apps

Skew : 3.72

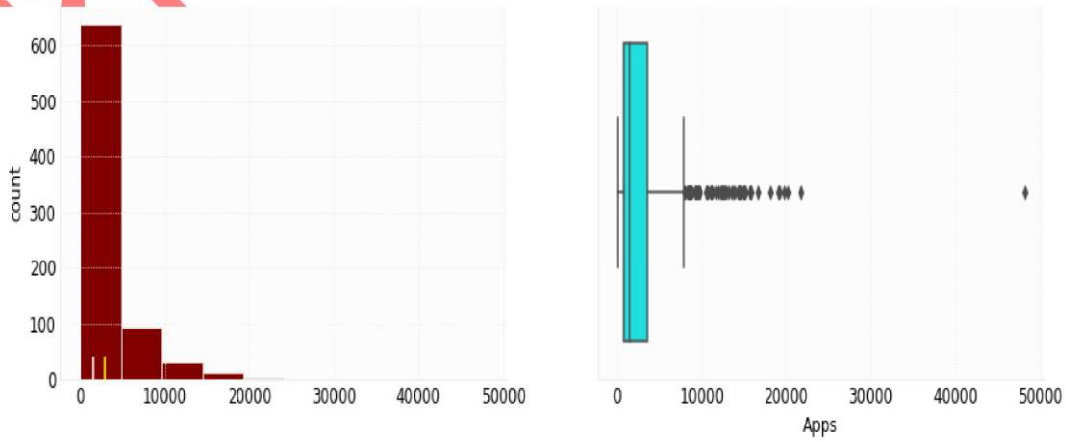



*Figure 2: Variable "Apps"*

Accept

Skew : 3.42

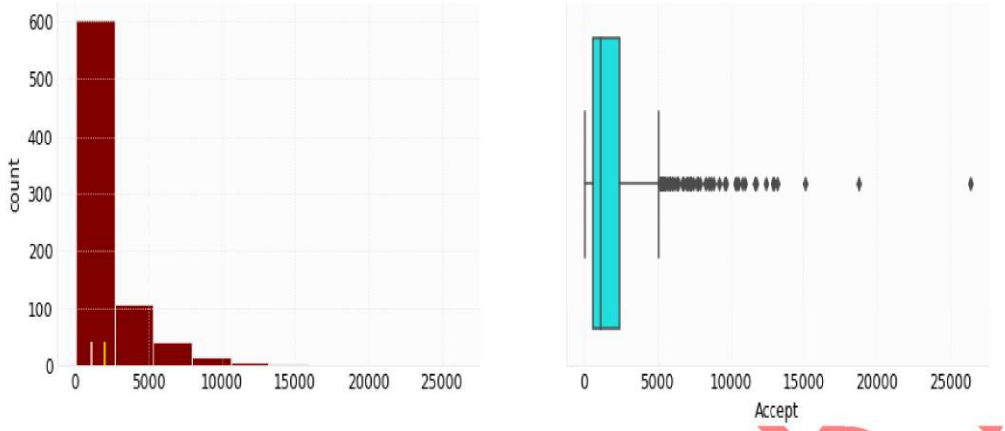



*Figure 3:Variable "Accept"*

Enroll

Skew : 2.69

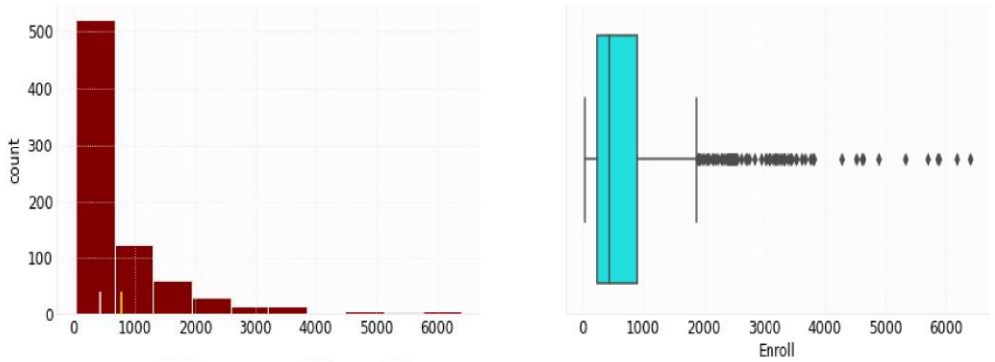



*Figure 4: Variable "Enroll"*

Top10perc

Skew : 1.41

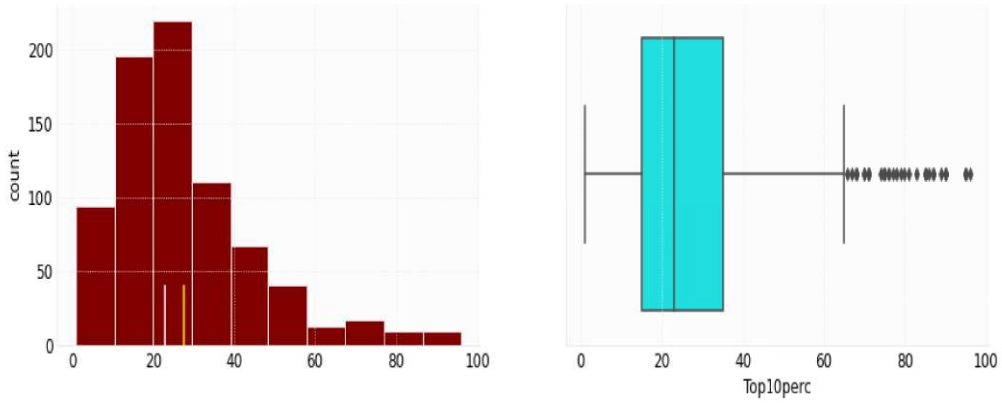



*Figure 5: Variable "Top10perc"*

Top25perc

Skew : 0.26

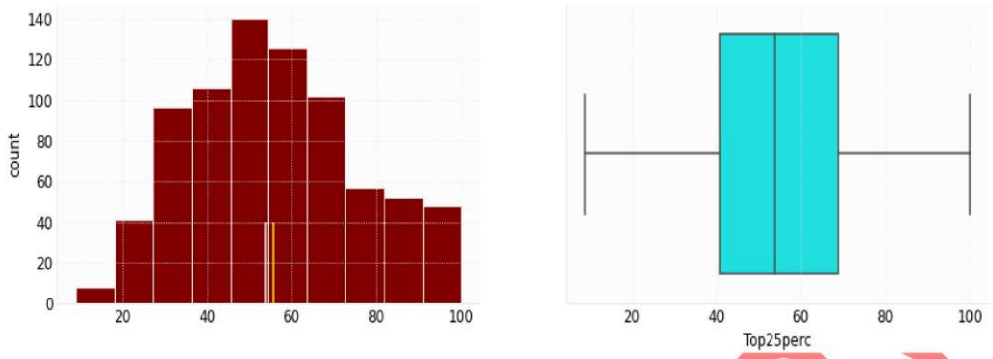



*Figure 6: Variable "Top25perc"*

F.Undergrad

Skew : 2.61

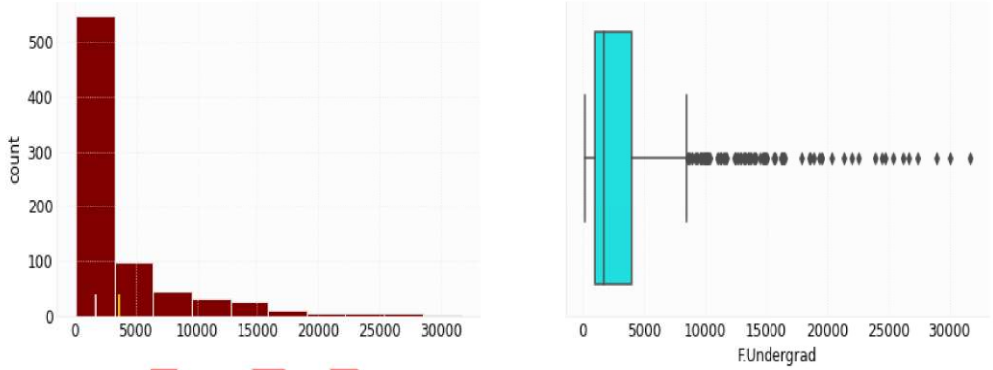



*Figure 7: Variable "F.Undergrad"*

P.Undergrad

Skew : 5.69

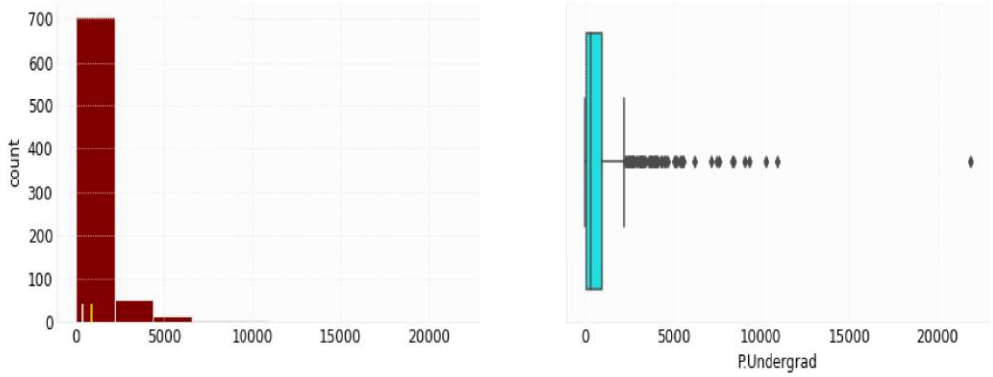



*Figure 8: Variable "P.Undergrad"*

Outstate

Skew : 0.51

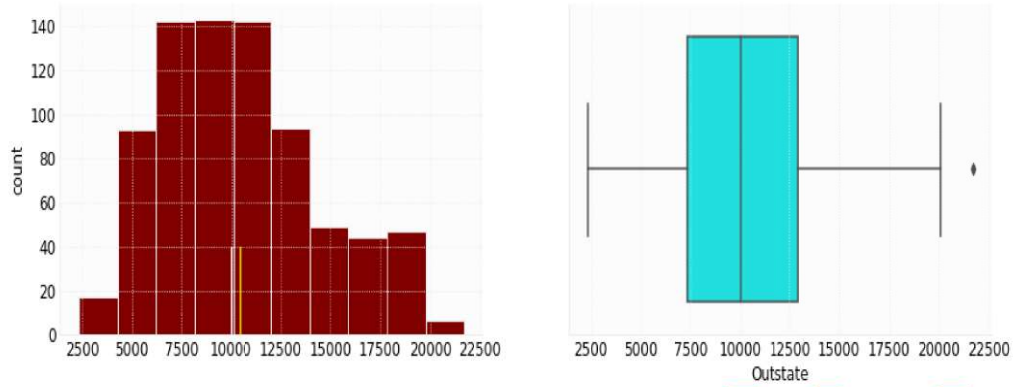



*Figure 9: Variable "Outstate"*

Room.Board

Skew : 0.48

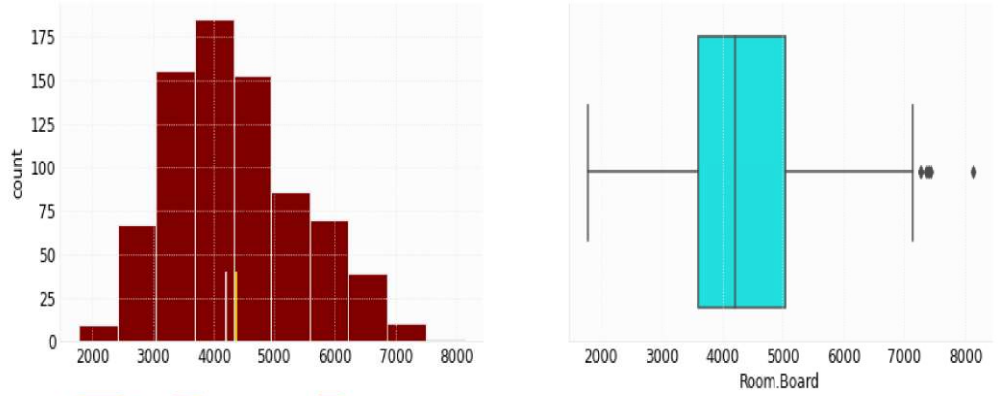



*Figure 10: Variable "Room.Board"*

Books

Skew : 3.49

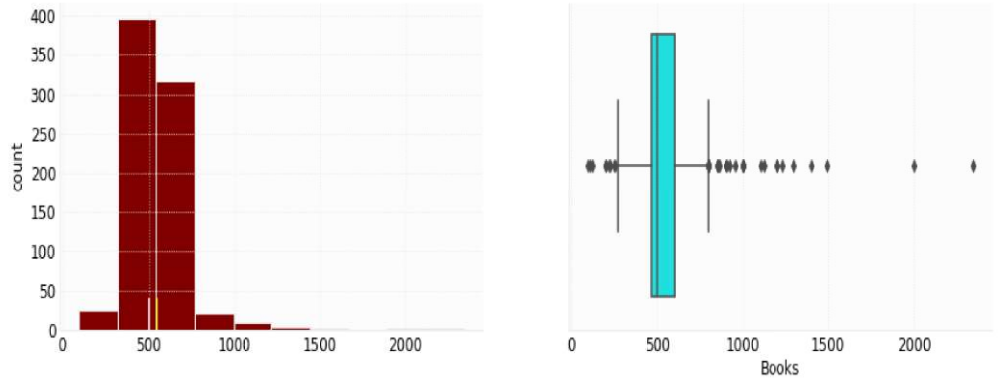



*Figure 11: Variable "Books"*

Personal

Skew : 1.74

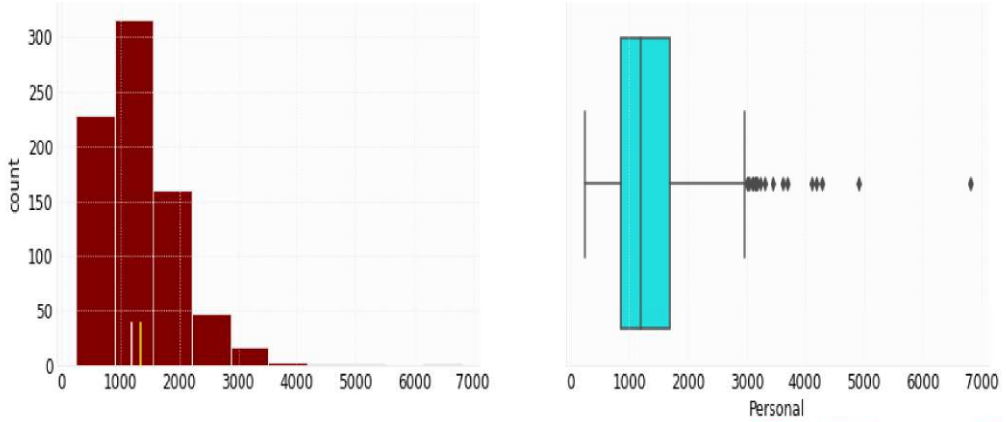



*Figure 12: Variable "Personal"*

PhD

Skew : -0.77

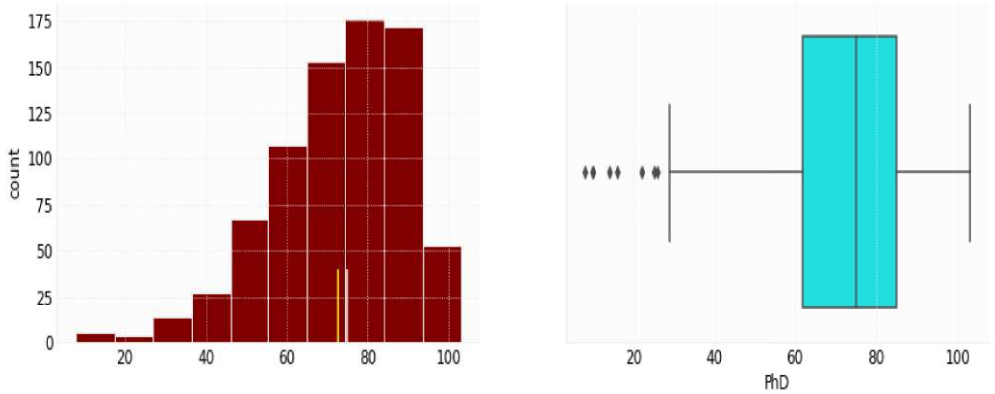



*Figure 13: Variable " PhD"*

Terminal

Skew : -0.82

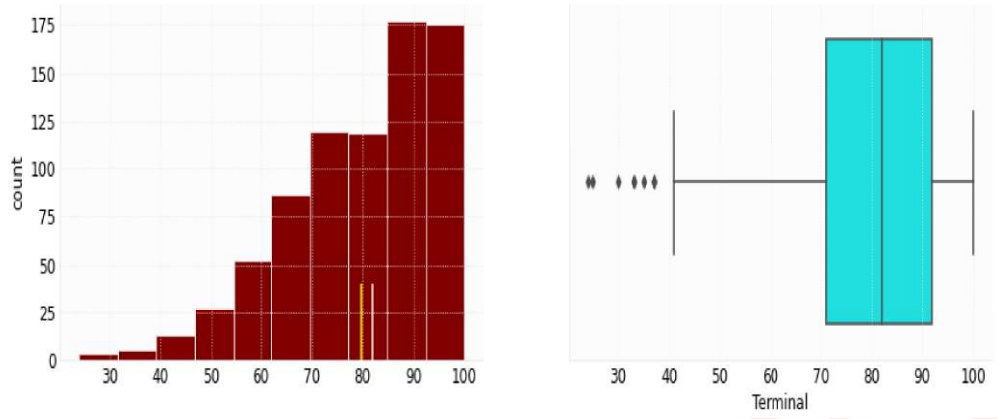



*Figure 14: Variable "Terminal"*

S.F.Ratio

Skew : 0.67

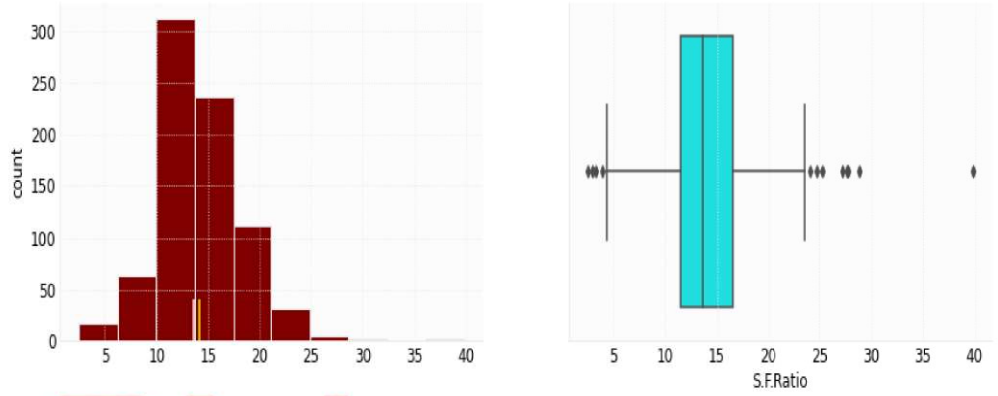



*Figure 15: Variable "S.F.Ratio"*

perc.alumni

Skew : 0.61

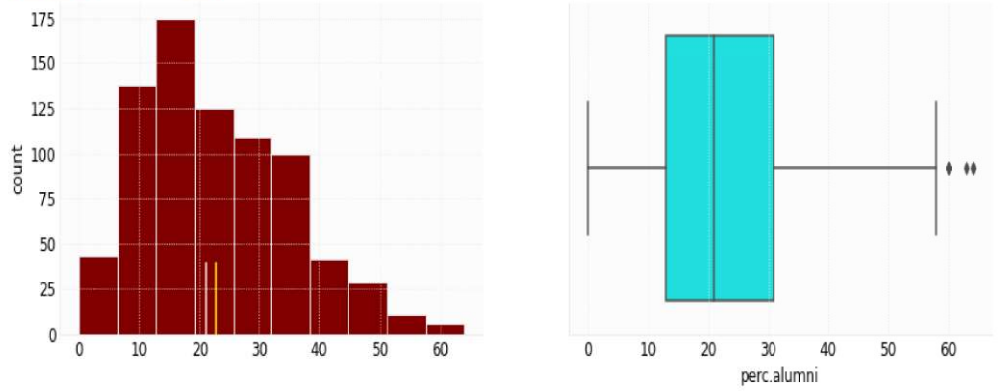



*Figure 16: Variable "perc.alumni"*

Expend

Skew : 3.46

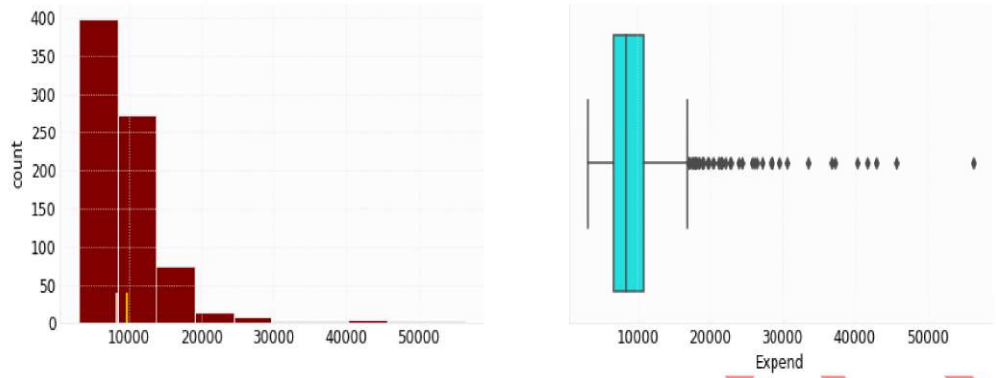



*Figure 17: Variable "Expend"*

Grad.Rate

Skew : -0.11

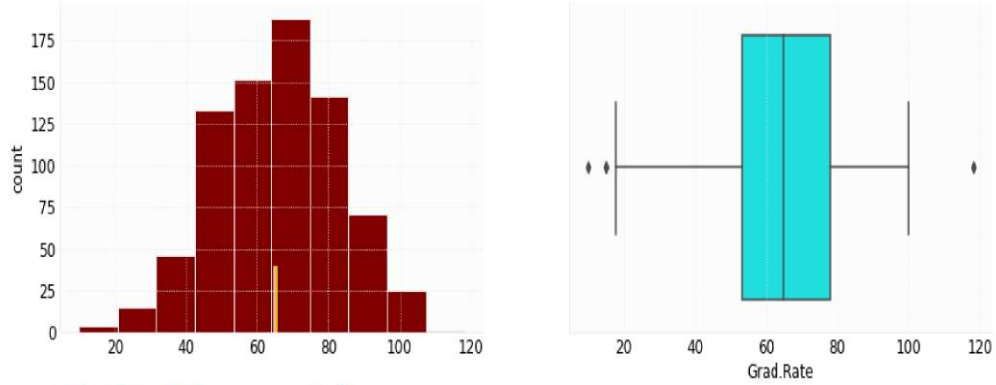



*Figure 18: Variable "Grad.Rate"*

The diagrams show that Apps, Accept, Enroll, Top10perc, F.Undergrad, P.Undergrad, Books, Personal and Expend variables are highly skewed. From the boxplot, it is evident that all these variables have outliers.

Top25percent is the only variable which does not possess outliers

Outstate, Room.Board, S.F.Ratio and perc.alumni seems to have a moderate right skew.

PhD and Terminal are the only variables which are left skewed but moderately.

## Bivariate Analysis

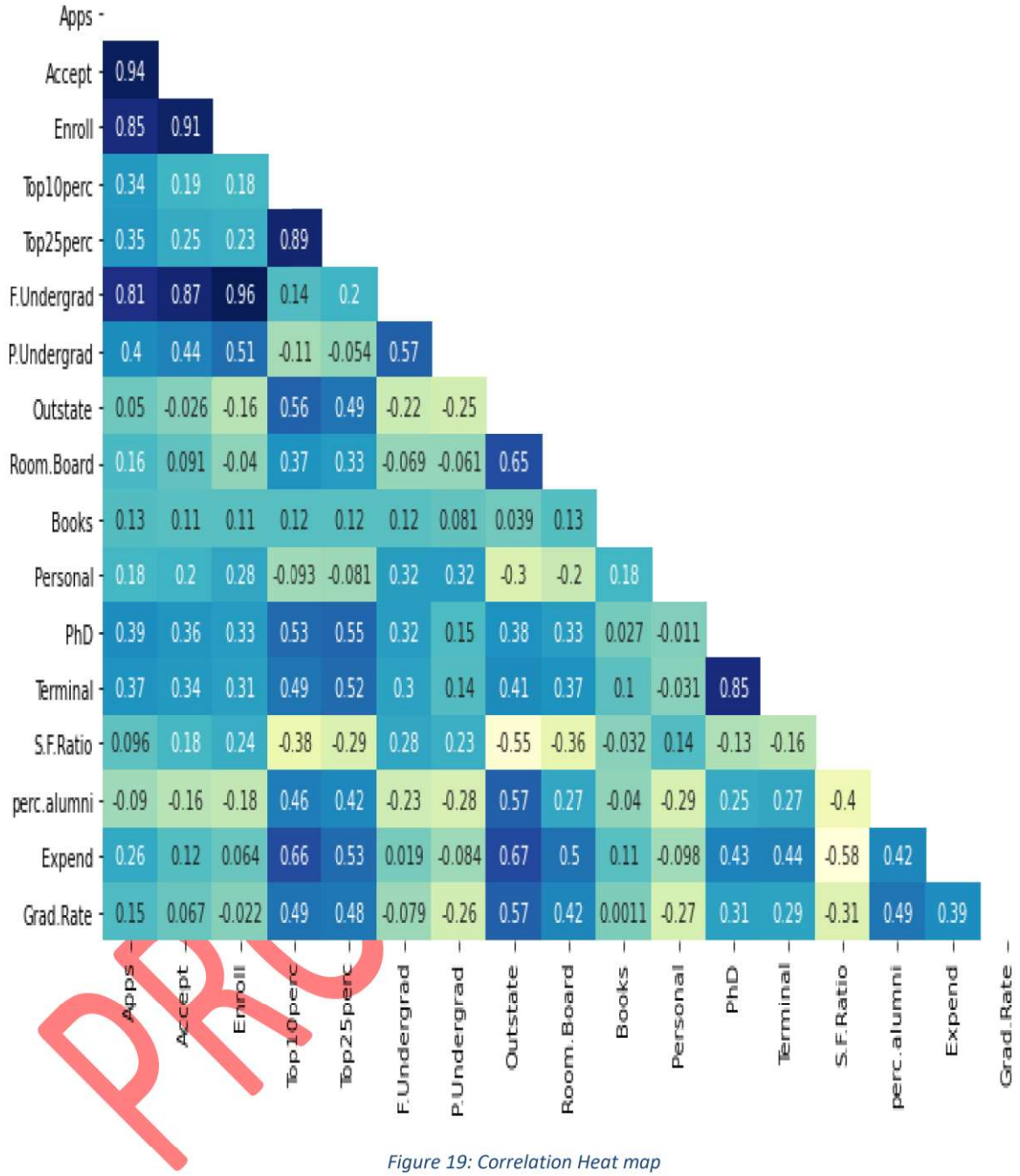



*Figure 19: Correlation Heat map*

We can see high positive correlation among following variables:

1. Apps and Accept
2. Apps and Enroll
3. Apps and F.Undergrad
4. Accept and Enroll
5. Accept and F.Undergrad
6. Enroll and F.Undergrad

7. Top10perc and Top25percent
8. PhD and Terminal

## Is scaling necessary for PCA in this case? Give justification and perform scaling.

PCA is impacted by scaling. If variables in the data set have large differences in their variances, then all variables need to be scaled. Otherwise the variables(s) with larges variance will have disproportionately more influence on the construction of PCs

| Feature | Variance Before Scaling | Variance After Scaling |
|---|---|---|
| Apps | 14978459.53 | 1 |
| Accept | 6007959.7 | 1 |
| Enroll | 863368.39 | 1 |
| Top10perc | 311.18 | 1 |
| Top25perc | 392.23 | 1 |
| F.Undergrad | 23526579.33 | 1 |
| P.Undergrad | 2317798.85 | 1 |
| Outstate | 16184661.63 | 1 |
| Room.Board | 1202743.03 | 1 |
| Books | 27259.78 | 1 |
| Personal | 458425.75 | 1 |
| PhD | 266.61 | 1 |
| Terminal | 216.75 | 1 |
| S.F.Ratio | 15.67 | 1 |
| perc.alumni | 153.56 | 1 |
| Expend | 27266865.64 | 1 |
| Grad.Rate | 295.07 | 1 |

Use StandardScaler to standardize the dataset's features onto unit scale (mean = 0 and variance = 1) which is a requirement for the optimal performance of many machine learning algorithms. The standard score of a sample x is calculated as:

$$z = \frac{(x - \bar{x})}{s}$$

*Equation 1: z score formula*

where u is the mean of the samples and s is the standard deviation of the sample. Centering and scaling happen independently on each feature by computing the relevant statistics on the samples in the dataset. Mean and standard deviation are then stored to be used on later data using transform.

We used zscore from scipy stats to standardise the data. Below is a transposed dataframe head.

| [redacted] | 0 | 1 | 2 | 3 | 4 |
|---|---|---|---|---|---|
| **Apps** | -0.35 | -0.21 | -0.41 | -0.67 | -0.73 |
| **Accept** | -0.32 | -0.04 | -0.38 | -0.68 | -0.76 |
| **Enroll** | -0.06 | -0.29 | -0.48 | -0.69 | -0.78 |
| **Top10perc** | -0.26 | -0.66 | -0.32 | 1.84 | -0.66 |
| **Top25perc** | -0.19 | -1.35 | -0.29 | 1.68 | -0.60 |
| **F.Undergrad** | -0.17 | -0.21 | -0.55 | -0.66 | -0.71 |
| **P.Undergrad** | -0.21 | 0.24 | -0.50 | -0.52 | 0.01 |
| **Outstate** | -0.75 | 0.46 | 0.20 | 0.63 | -0.72 |
| **Room.Board** | -0.96 | 1.91 | -0.55 | 1.00 | -0.22 |
| **Books** | -0.60 | 1.22 | -0.91 | -0.60 | 1.52 |
| **Personal** | 1.27 | 0.24 | -0.26 | -0.69 | 0.24 |
| **PhD** | -0.16 | -2.68 | -1.20 | 1.19 | 0.20 |
| **Terminal** | -0.12 | -3.38 | -0.93 | 1.18 | -0.52 |
| **S.F.Ratio** | 1.01 | -0.48 | -0.30 | -1.62 | -0.55 |
| **perc.alumni** | -0.87 | -0.54 | 0.59 | 1.15 | -1.68 |
| **Expend** | -0.50 | 0.17 | -0.18 | 1.79 | 0.24 |
| **Grad.Rate** | -0.32 | -0.55 | -0.67 | -0.38 | -2.94 |

As we can see from the above df, data after scaling will transform every value in such a way that the mean will be 0 and standard deviation will be 1.

Comment on the comparison between the covariance and the correlation matrices from this data.

From the below images, it is clear is that the correlation and the covariance matrices are same after z-score (mean =0 and sd=1) scaling is performed.

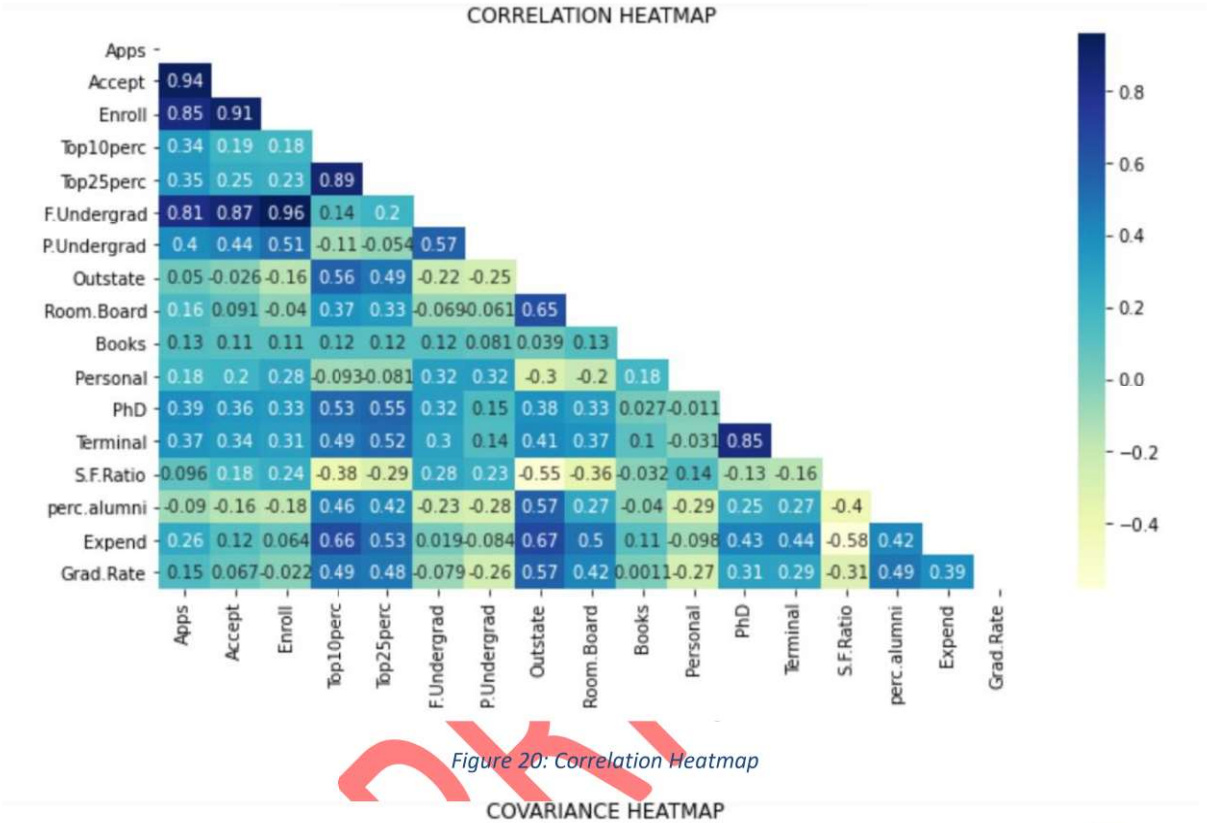



*Figure 20: Correlation Heatmap*

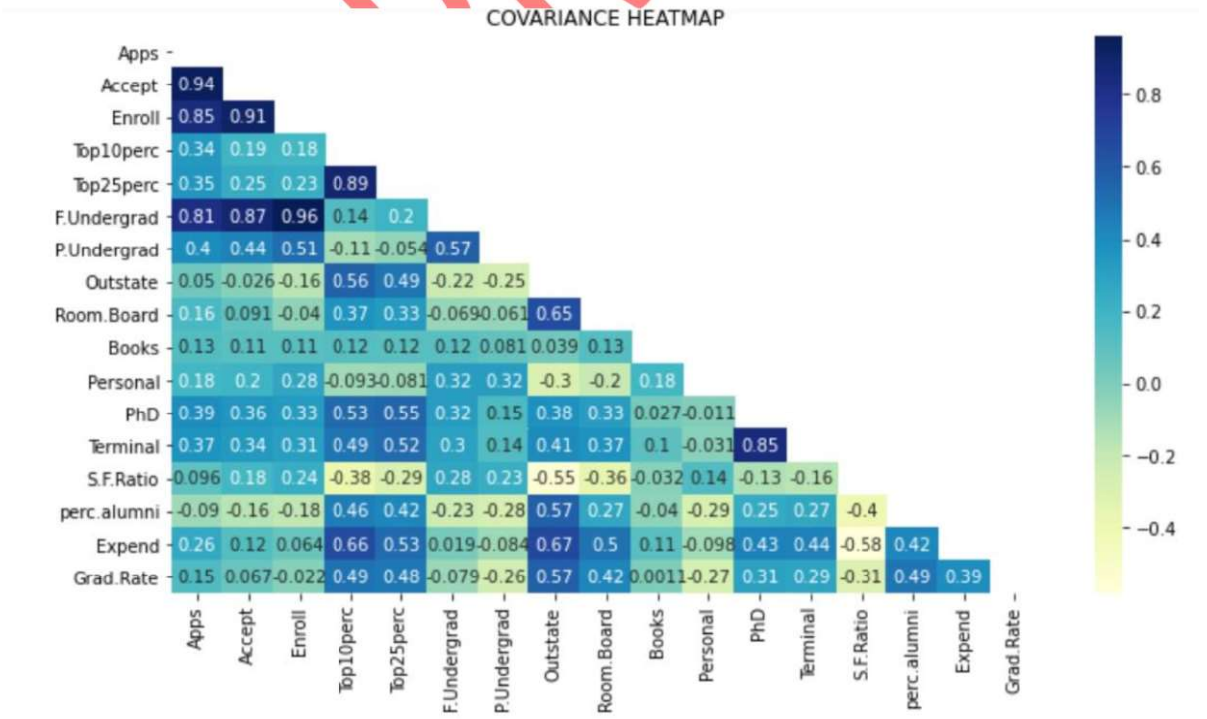
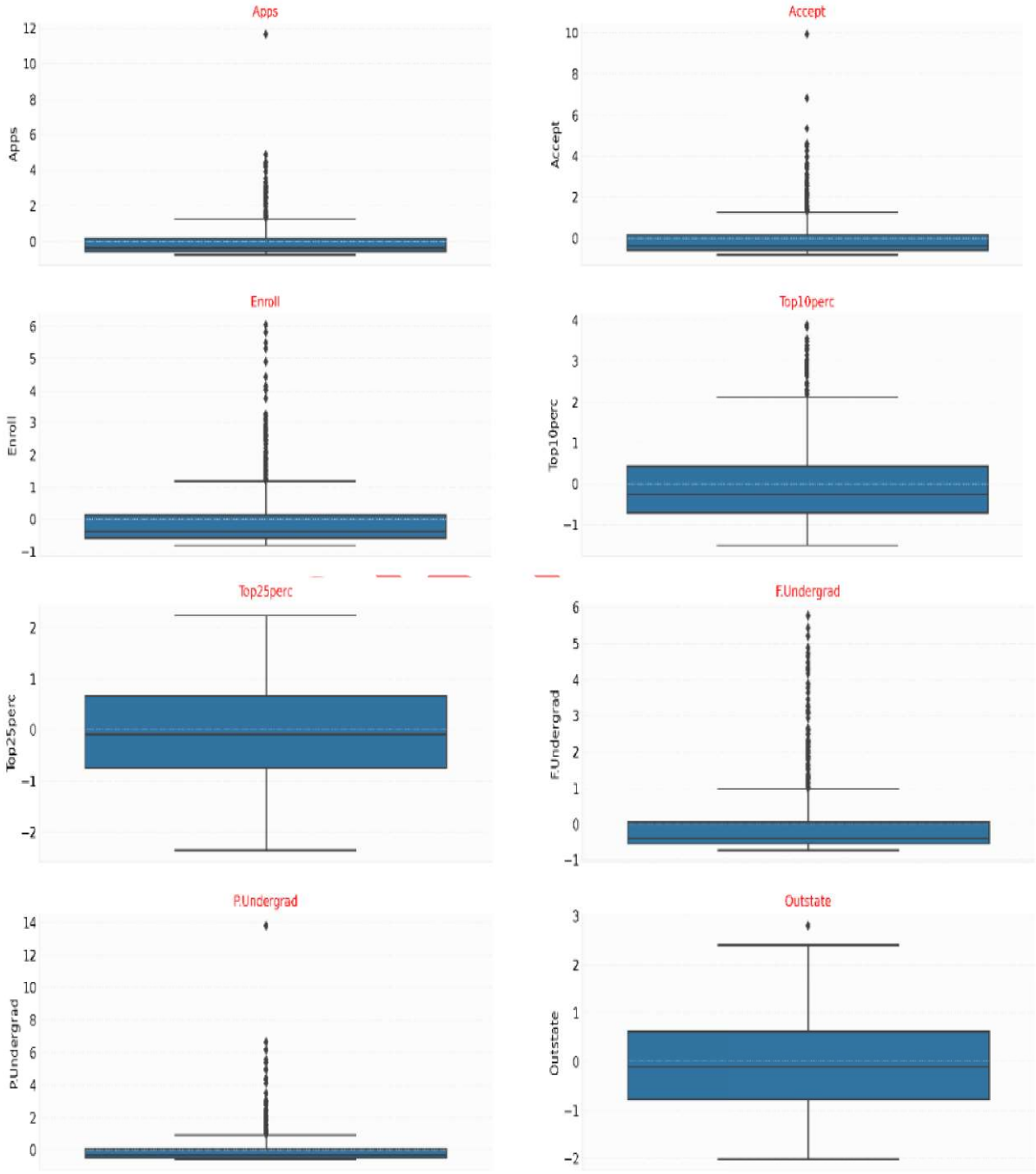



*Figure 21: Covariance Heatmap*

Check the dataset for outliers before and after scaling. What insight do you derive here? **[No need to treat the outliers]**

Scaling of data does not have any impact on outliers. The primary purpose of scaling is to make sure all variable is on the same scale, but that does not have any effect on the existing outliers. Similar to Q1 - univariate analysis boxplots, we can see that same outliers exist even after scaling the data.



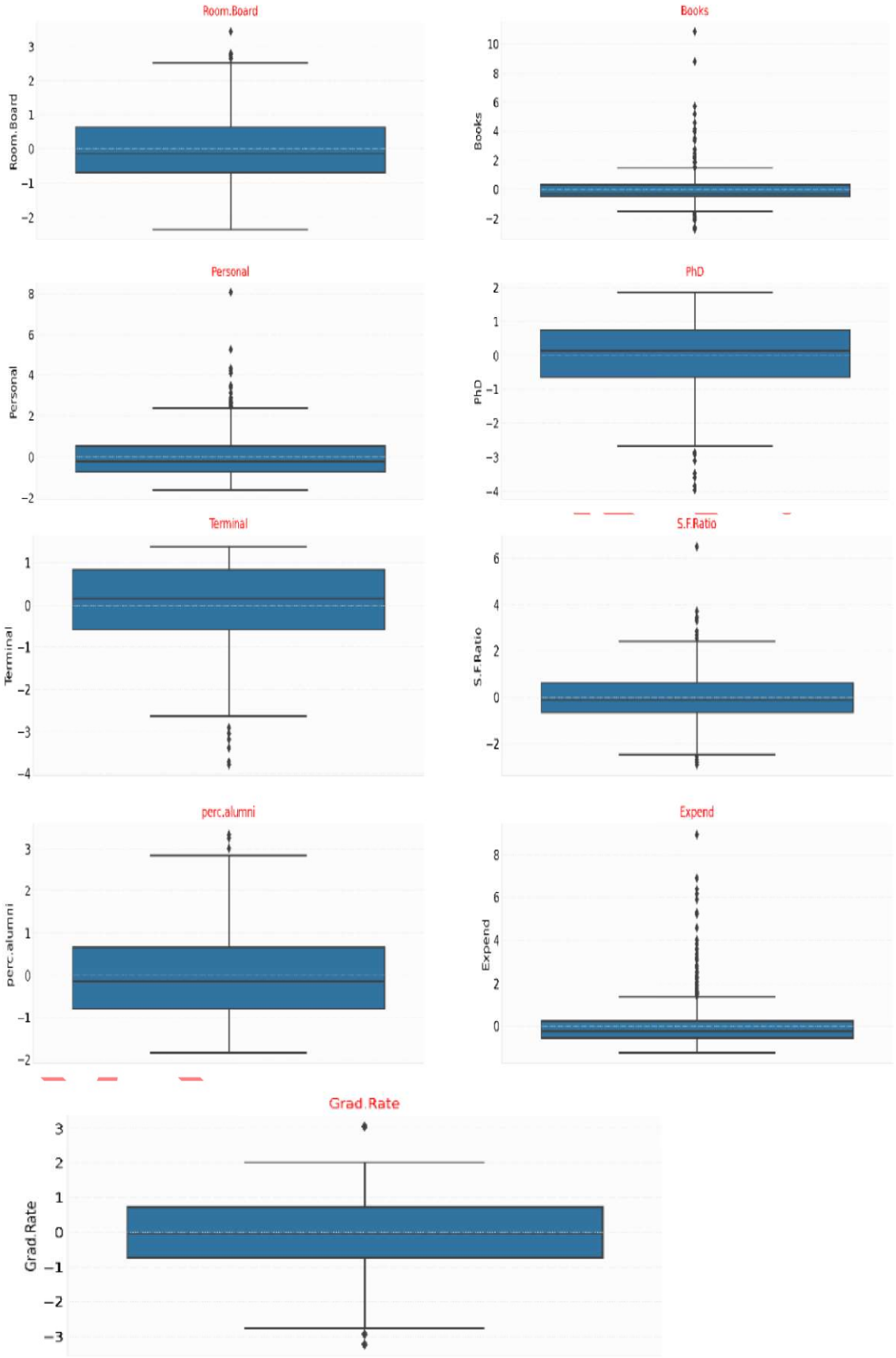



*Figure 22: Outliers check using Boxplot*

Extract the eigenvalues, and eigenvectors.

Extracted using linear algebra from Numpy. Eigen values are:

5.44, 4.48, 1.17, 1.01, 0.93, 0.85, 0.61, 0.59, 0.53, 0.4, 0.02, 0.04, 0.31, 0.09, 0.14, 0.17, 0.22

Eigen vectors are:

```
[[-0.25,  0.33,  0.06, -0.28,  0.01,  0.02,  0.04,  0.1 ,  0.09,
       -0.05,  0.36, -0.46,  0.04, -0.13,  0.08, -0.6 ,  0.02],
      [-0.21,  0.37,  0.1 , -0.27,  0.06, -0.01,  0.01,  0.06,  0.18,
       -0.04, -0.54,  0.52, -0.06,  0.15,  0.03, -0.29, -0.15],
      [-0.18,  0.4 ,  0.08, -0.16, -0.06,  0.04,  0.03, -0.06,  0.13,
       -0.03,  0.61,  0.4 , -0.07, -0.03, -0.09,  0.44,  0.01],
      [-0.35, -0.08, -0.04,  0.05, -0.4 ,  0.05,  0.16,  0.12, -0.34,
       -0.06, -0.14,  0.15, -0.01, -0.7 , -0.11, -0.  ,  0.04],
      [-0.34, -0.04,  0.02,  0.11, -0.43, -0.03,  0.12,  0.1 , -0.4 ,
       -0.01,  0.08, -0.05, -0.27,  0.62,  0.15, -0.02, -0.09],
      [-0.15,  0.42,  0.06, -0.1 , -0.04,  0.04,  0.03, -0.08,  0.06,
       -0.02, -0.41, -0.56, -0.08, -0.01, -0.06,  0.52,  0.06],
      [-0.03,  0.32, -0.14,  0.16,  0.3 ,  0.19, -0.06, -0.57, -0.56,
        0.22,  0.01,  0.05,  0.1 , -0.02,  0.02, -0.13, -0.06],
      [-0.29, -0.25, -0.05, -0.13,  0.22,  0.03, -0.11, -0.01,  0.  ,
       -0.19,  0.05, -0.1 ,  0.14, -0.04, -0.03,  0.14, -0.82],
      [-0.25, -0.14, -0.15, -0.18,  0.56, -0.16, -0.21,  0.22, -0.28,
       -0.3 ,  0.  ,  0.03, -0.36, -0.  , -0.06,  0.07,  0.35],
      [-0.06,  0.06, -0.68, -0.09, -0.13, -0.64,  0.15, -0.21,  0.13,
        0.08,  0.  , -0.  ,  0.03,  0.01, -0.07, -0.01, -0.03],
      [ 0.04,  0.22, -0.5 ,  0.23, -0.22,  0.33, -0.63,  0.23,  0.09,
       -0.14, -0.  ,  0.01, -0.02,  0.  ,  0.03, -0.04, -0.04],
      [-0.32,  0.06,  0.13,  0.53,  0.14, -0.09,  0.  ,  0.08,  0.19,
        0.12,  0.01, -0.03,  0.04,  0.11, -0.69, -0.13,  0.02],
```

```
[-0.32,  0.05,  0.07,  0.52,  0.2 , -0.15,  0.03,  0.01,  0.25,
  0.09,  0.01,  0.03, -0.06, -0.16,  0.67,  0.06,  0.02],
[ 0.18,  0.25,  0.29,  0.16, -0.08, -0.49, -0.22,  0.08, -0.27,
 -0.47, -0.  ,  0.02,  0.45,  0.02,  0.04,  0.02, -0.01],
[-0.21, -0.25,  0.15, -0.02, -0.22,  0.05, -0.24, -0.68,  0.26,
 -0.42, -0.02, -0.  , -0.13,  0.01, -0.03, -0.1 ,  0.18],
[-0.32, -0.13, -0.23, -0.08,  0.08,  0.3 ,  0.23,  0.05,  0.05,
 -0.13, -0.04,  0.04,  0.69,  0.23,  0.07,  0.09,  0.33],
[-0.25, -0.17,  0.21, -0.27, -0.11, -0.22, -0.56,  0.01, -0.04,
  0.59, -0.01,  0.01,  0.22,  0.  ,  0.04,  0.07,  0.12]]
```

## Perform PCA and export the data of the Principal Component (eigenvectors) into a data frame with the original features.

PCA has been performed using Numpy's linear algebra to calculate eigenvectors and eigenvalues from covariance matrix

| | PC1 | PC2 | PC3 | PC4 | PC5 | PC6 | PC7 | PC8 |
|---|---|---|---|---|---|---|---|---|
| **Apps** | -0.25 | 0.33 | 0.06 | -0.28 | 0.01 | 0.02 | 0.04 | 0.10 |
| **Accept** | -0.21 | 0.37 | 0.10 | -0.27 | 0.06 | -0.01 | 0.01 | 0.06 |
| **Enroll** | -0.18 | 0.40 | 0.08 | -0.16 | -0.06 | 0.04 | 0.03 | -0.06 |
| **Top10perc** | -0.35 | -0.08 | -0.04 | 0.05 | -0.40 | 0.05 | 0.16 | 0.12 |
| **Top25perc** | -0.34 | -0.04 | 0.02 | 0.11 | -0.43 | -0.03 | 0.12 | 0.10 |
| **F.Undergrad** | -0.15 | 0.42 | 0.06 | -0.10 | -0.04 | 0.04 | 0.03 | -0.08 |

| | PC1 | PC2 | PC3 | PC4 | PC5 | PC6 | PC7 | PC8 |
|---|---|---|---|---|---|---|---|---|
| **P.Undergrad** | -0.03 | 0.32 | -0.14 | 0.16 | 0.30 | 0.19 | -0.06 | -0.57 |
| **Outstate** | -0.29 | -0.25 | -0.05 | -0.13 | 0.22 | 0.03 | -0.11 | -0.01 |
| **Room.Board** | -0.25 | -0.14 | -0.15 | -0.18 | 0.56 | -0.16 | -0.21 | 0.22 |
| **Books** | -0.06 | 0.06 | -0.68 | -0.09 | -0.13 | -0.64 | 0.15 | -0.21 |
| **Personal** | 0.04 | 0.22 | -0.50 | 0.23 | -0.22 | 0.33 | -0.63 | 0.23 |
| **PhD** | -0.32 | 0.06 | 0.13 | 0.53 | 0.14 | -0.09 | 0.00 | 0.08 |
| **Terminal** | -0.32 | 0.05 | 0.07 | 0.52 | 0.20 | -0.15 | 0.03 | 0.01 |
| **S.F.Ratio** | 0.18 | 0.25 | 0.29 | 0.16 | -0.08 | -0.49 | -0.22 | 0.08 |
| **perc.alumni** | -0.21 | -0.25 | 0.15 | -0.02 | -0.22 | 0.05 | -0.24 | -0.68 |
| **Expend** | -0.32 | -0.13 | -0.23 | -0.08 | 0.08 | 0.30 | 0.23 | 0.05 |
| **Grad.Rate** | -0.25 | -0.17 | 0.21 | -0.27 | -0.11 | -0.22 | -0.56 | 0.01 |

| | PC9 | PC10 | PC11 | PC12 | PC13 | PC14 | PC15 | PC16 | PC17 |
|---|---|---|---|---|---|---|---|---|---|
| **Apps** | 0.09 | -0.05 | 0.36 | -0.46 | 0.04 | -0.13 | 0.08 | -0.60 | 0.02 |
| **Accept** | 0.18 | -0.04 | -0.54 | 0.52 | -0.06 | 0.15 | 0.03 | -0.29 | -0.15 |

| | PC9 | PC10 | PC11 | PC12 | PC13 | PC14 | PC15 | PC16 | PC17 |
|---|---|---|---|---|---|---|---|---|---|
| **Enroll** | 0.13 | -0.03 | 0.61 | 0.40 | -0.07 | -0.03 | -0.09 | 0.44 | 0.01 |
| **Top10perc** | -0.34 | -0.06 | -0.14 | 0.15 | -0.01 | -0.70 | -0.11 | -0.00 | 0.04 |
| **Top25perc** | -0.40 | -0.01 | 0.08 | -0.05 | -0.27 | 0.62 | 0.15 | -0.02 | -0.09 |
| **F.Undergrad** | 0.06 | -0.02 | -0.41 | -0.56 | -0.08 | -0.01 | -0.06 | 0.52 | 0.06 |
| **P.Undergrad** | -0.56 | 0.22 | 0.01 | 0.05 | 0.10 | -0.02 | 0.02 | -0.13 | -0.06 |
| **Outstate** | 0.00 | -0.19 | 0.05 | -0.10 | 0.14 | -0.04 | -0.03 | 0.14 | -0.82 |
| **Room.Board** | -0.28 | -0.30 | 0.00 | 0.03 | -0.36 | -0.00 | -0.06 | 0.07 | 0.35 |
| **Books** | 0.13 | 0.08 | 0.00 | -0.00 | 0.03 | 0.01 | -0.07 | -0.01 | -0.03 |
| **Personal** | 0.09 | -0.14 | -0.00 | 0.01 | -0.02 | 0.00 | 0.03 | -0.04 | -0.04 |
| **PhD** | 0.19 | 0.12 | 0.01 | -0.03 | 0.04 | 0.11 | -0.69 | -0.13 | 0.02 |
| **Terminal** | 0.25 | 0.09 | 0.01 | 0.03 | -0.06 | -0.16 | 0.67 | 0.06 | 0.02 |
| **S.F.Ratio** | -0.27 | -0.47 | -0.00 | 0.02 | 0.45 | 0.02 | 0.04 | 0.02 | -0.01 |
| **perc.alumni** | 0.26 | -0.42 | -0.02 | -0.00 | -0.13 | 0.01 | -0.03 | -0.10 | 0.18 |
| **Expend** | 0.05 | -0.13 | -0.04 | 0.04 | 0.69 | 0.23 | 0.07 | 0.09 | 0.33 |

| | PC9 | PC10 | PC11 | PC12 | PC13 | PC14 | PC15 | PC16 | PC17 |
|---|---|---|---|---|---|---|---|---|---|
| **Grad.Rate** | -0.04 | 0.59 | -0.01 | 0.01 | 0.22 | 0.00 | 0.04 | 0.07 | 0.12 |

Write down the explicit form of the first PC (in terms of the eigenvectors. Use values with two places of decimals only).

The equation for first PC would be the cross multiplication of variables and the Eigen vectors of 0 index. Eigen vectors derived from sklearn, Numpy and statsmodels might differ a bit numerically or on the sign perspective.

```
Apps          -0.25
Accept        -0.21
Enroll        -0.18
Top10perc     -0.35
Top25perc     -0.34
F.Undergrad   -0.15
P.Undergrad   -0.03
Outstate      -0.29
Room.Board    -0.25
Books         -0.06
Personal       0.04
PhD           -0.32
Terminal      -0.32
S.F.Ratio      0.18
perc.alumni   -0.21
Expend        -0.32
Grad.Rate     -0.25
```

***PC1: (-0.25 x S_Apps) + (-0.21 x S_Accept) + (-0.18 x S_Enroll) + (-0.35 x S_Top10perc) + (-0.34 x S_Top25perc) + (-0.15 x S_F.Undergrad) + (-0.03 x S_P.Undergrad) + (-0.29 x S_Outstate) + (-0.25 x S_Room.Board) + (-0.06 x S_Books) + (-0.04 x S_Personal) + (-0.32 x S_PhD) + (-0.32 x S_Terminal) + (0.18 x S_S.F.Ratio) + (-0.21 x S_perc.alumni) + (-0.32 x S_Expend) + (-0.25 x S_Grad.Rate)***

*Equation 2: Explicit form of PC1*

Consider the cumulative values of the eigenvalues. How does it help you to decide on the optimum number of principal components? What do the eigenvectors indicate?

Explained variance by each principal component:

```
[32.02 26.34  6.9   5.92  5.49  4.98  3.56  3.45  3.12  2.38  1.84  1.3
 0.99  0.85  0.52  0.22  0.14]
```

The first PC explain 32% of variance in the dataset followed by 26.34% of variance explained by PC2 and so on.

Cumulative explained variance :

```
[ 32.02  58.36  65.26  71.18  76.67  81.66  85.22  88.67  91.79  94.16
 96.    97.3   98.29  99.13  99.65  99.86 100.  ]
```

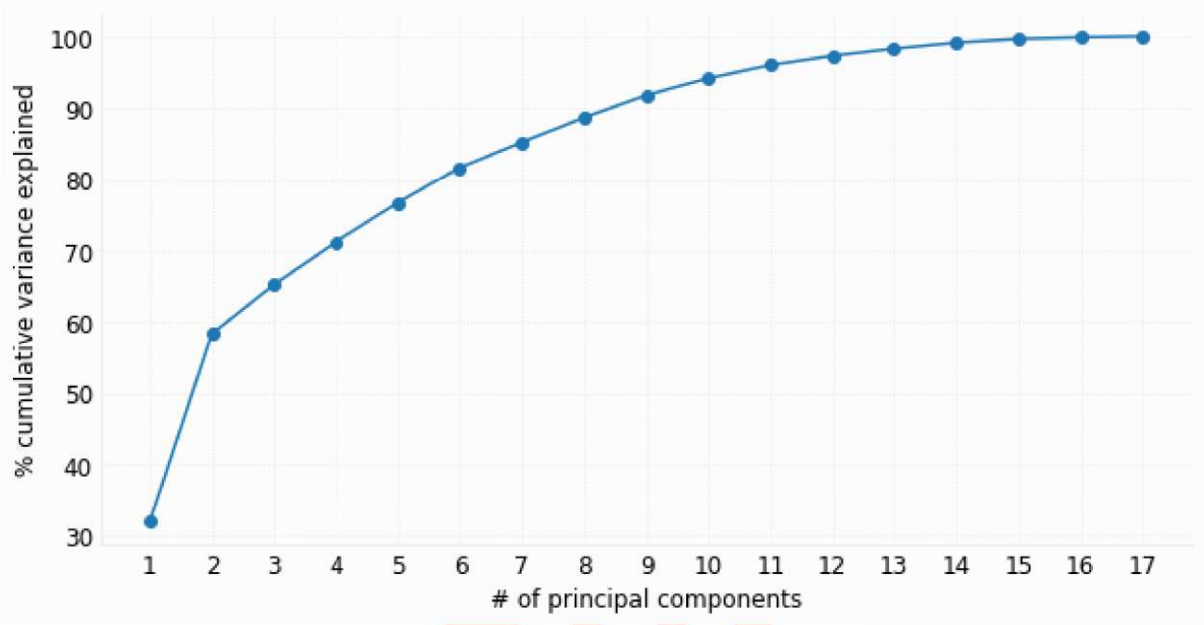



*Figure 23: Cumulative % of variance by PCs*

The Cumulative % gives the percentage of variance accounted for by the n components. For example, the cumulative percentage for the second component is the sum of the percentage of variance for the first and second components. It helps in deciding the number of components by selecting the components which explained the high variance.

In the above array we see that the first feature explains 32% of the variance within our data set while the first two explains 58.3 and so on. If we employ 13 features we capture 98.3% of the variance within the dataset, thus we gain very little by implementing an additional feature (think of this as diminishing marginal return on total variance explained).

Explain the business implication of using the Principal Component Analysis for this case study. How may PCs help in further analysis? [**Hint:** Write Interpretations of the Principal Components Obtained]

Interpretation of the principal component is based on which variables are most strongly correlated with each component.

Example Component Summary:

***PC1: (-0.25 x S_Apps) + (-0.21 x S_Accept) + (-0.18 x S_Enroll) + (-0.35 x S_Top10perc) + (-0.34 x S_Top25perc) + (-0.15 x S_F.Undergrad) + (-0.03 x S_P.Undergrad) + (-0.29 x S_Outstate) + (-0.25 x S_Room.Board) + (-0.06 x S_Books) + (-0.04 x S_Personal) + (-0.32 x S_PhD) + (-0.32 x S_Terminal) + (0.18 x S_S.F.Ratio) + (-0.21 x S_perc.alumni) + (-0.32 x S_Expend) + (-0.25 x S_Grad.Rate)***

*Equation 3: Explicit form of PC1*

The letter S indicates that the scaled (normalized) variable is used to construct the PCs.

Similarly, the other PCs can also be expressed in terms of the scaled variables.

For the business implications, the following can be a way to explain this Principal Component Analysis:
Let's consider the correlation values b/w the PCs and the Original Features (see Table 1)

- The first principal component picks up around 32% of the variability in the data. That is to say, picking up a considerable amount of variation in the data. So in a business scenario, we need to look at the ease of doing things as well.
- The explanation of each component along with their weights is also one of the ways to look at it.

*Table 1: Principal Components with Features*

| | PC1 | PC2 | PC3 | PC4 |
|---|---|---|---|---|
| **Apps** | 0.250000 | 0.330000 | -0.060000 | 0.280000 |
| **Accept** | 0.210000 | 0.370000 | -0.100000 | 0.270000 |
| **Enroll** | 0.180000 | 0.400000 | -0.080000 | 0.160000 |
| **Top10perc** | 0.350000 | -0.080000 | 0.040000 | -0.050000 |
| **Top25perc** | 0.340000 | -0.040000 | -0.020000 | -0.110000 |
| **F.Undergrad** | 0.150000 | 0.420000 | -0.060000 | 0.100000 |
| **P.Undergrad** | 0.030000 | 0.320000 | 0.140000 | -0.160000 |
| **Outstate** | 0.290000 | -0.250000 | 0.050000 | 0.130000 |

| | PC1 | PC2 | PC3 | PC4 |
|---|---|---|---|---|
| **Room.Board** | 0.250000 | -0.140000 | 0.150000 | 0.180000 |
| **Books** | 0.060000 | 0.060000 | 0.680000 | 0.090000 |
| **Personal** | -0.040000 | 0.220000 | 0.500000 | -0.230000 |
| **PhD** | 0.320000 | 0.060000 | -0.130000 | -0.530000 |
| **Terminal** | 0.320000 | 0.050000 | -0.070000 | -0.520000 |
| **S.F.Ratio** | -0.180000 | 0.250000 | -0.290000 | -0.160000 |
| **perc.alumni** | 0.210000 | -0.250000 | -0.150000 | 0.020000 |
| **Expend** | 0.320000 | -0.130000 | 0.230000 | 0.080000 |
| **Grad.Rate** | 0.250000 | -0.170000 | -0.210000 | 0.270000 |

**Component Summaries**

- First Principal Component - PC1
  - The first principal component is a measure of the quality of Apps,Top10perc, Top25perc,Terminal,and PhD. This component is associated with high scores on all of these variables. They are all positively related to PC1 since they all have positive signs.
- Second Principal Component - PC2
  - The second principal component is a measure of the accept, enrollment, F.Undergrad,P.Undergrad,Outstate,perc.alumni. PC2 is associated with high scores of F.Undergrad and Enrollment and low ratings of alumni donation & Outstation students.
- Third Principal Component - PC3
  - The third principal component is a measure of the Books' cost and Personal Spending of the student.
- Fourth Principal Component - PC4
  - The fourth principal component is a measure of the Cost of Boarding, Personal Spending, % of faculties with PhD or terminal degree. PC4 is associated with

high scores of Cost of Boarding with low ratings of Personal Spending and % of faculties with PhD or terminal degrees.

***Note**: Signs will not matter in the interpretation but the magnitude.*

**Decision regarding which correlation value is high may vary from case to case. In this example, we have taken 25-30% to be of a considerable magnitude irrespective of the sign.**

**Principal components analysis is a very versatile technique and has its application in a number of situations.**

# Appendix

## Python Code

### Problem 1A:

In [1]:
```
import numpy as np
import pandas as pd
from scipy import stats
import seaborn as sns
import matplotlib.pyplot as plt
```

In [2]:
```
df = pd.read_csv('SalaryData.csv')
```

In [3]:
```
df.head()
```

Out[3]:

| | Serial No | Education | Occupation | Salary |
|---|---|---|---|---|
| 0 | 1 | Doctorate | Adm-clerical | 153197 |
| 1 | 2 | Doctorate | Adm-clerical | 115945 |
| 2 | 3 | Doctorate | Adm-clerical | 175935 |
| 3 | 4 | Doctorate | Adm-clerical | 220754 |
| 4 | 5 | Doctorate | Sales | 170769 |

In [4]:
```
salary_doctorate = df.loc[(df['Education'] == 'Doctorate'),'Salary']
salary_bachelors = df.loc[(df['Education'] == 'Bachelors'),'Salary']
salary_hsgrad = df.loc[(df['Education'] == 'HS-grad'),'Salary']
```

In [5]:
```
print(stats.f_oneway(salary_doctorate,salary_bachelors,salary_hsgrad))
```

```
F_onewayResult(statistic=30.95628008792558, pvalue=1.2577090926629002e-08)
```

In [6]:
```
#Level of Significance α = 0.05
salary_prof = df.loc[(df['Occupation'] == 'Prof-specialty'),'Salary']
salary_sales = df.loc[(df['Occupation'] == 'Sales'),'Salary']
salary_adm_cler = df.loc[(df['Occupation'] == 'Adm-clerical'),'Salary']
salary_exec_mngr = df.loc[(df['Occupation'] == 'Exec-managerial'),'Salary']
```

In [7]:
```
print(stats.f_oneway(salary_prof,salary_sales,salary_adm_cler,salary_exec_mngr))
```

```
F_onewayResult(statistic=0.8841441289216039, pvalue=0.4585078266495116)
```

In [8]:
```
from statsmodels.stats.multicomp import pairwise_tukeyhsd
m_comp = pairwise_tukeyhsd(endog=df['Salary'], groups=df['Education'], alpha=0.05)
m_comp.summary()
```

Out[8]:

Multiple Comparison of Means - Tukey HSD, FWER=0.05

| group1 | group2 | meandiff | p-adj | lower | upper | reject |
|---|---|---|---|---|---|---|
| Bachelors | Doctorate | 43274.0667 | 0.0146 | 7541.1439 | 79006.9894 | True |
| Bachelors | HS-grad | -90114.1556 | 0.001 | -132035.1958 | -48193.1153 | True |
| Doctorate | HS-grad | -133388.2222 | 0.001 | -174815.0876 | -91961.3569 | True |

From the output it is clear that mean salary is significantly different for each pair of means.

## Problem 1B:

In [1]:
```
import numpy as np
import pandas as pd
from scipy import stats
import statsmodels.api as sm
from statsmodels.formula.api import ols
import seaborn as sns
import matplotlib.pyplot as plt
from statsmodels.graphics.factorplots import interaction_plot
```

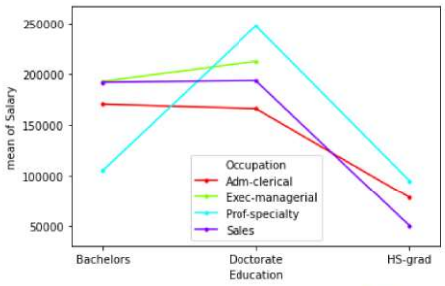
In [2]:
```
df = pd.read_csv('SalaryData.csv')
```

In [3]:
```
df.head()
```

Out[3]:

| | Serial No | Education | Occupation | Salary |
|---|---|---|---|---|
| 0 | 1 | Doctorate | Adm-clerical | 153197 |
| 1 | 2 | Doctorate | Adm-clerical | 115945 |
| 2 | 3 | Doctorate | Adm-clerical | 175935 |
| 3 | 4 | Doctorate | Adm-clerical | 220754 |
| 4 | 5 | Doctorate | Sales | 170769 |

In [4]:
```
fig = interaction_plot(x = df['Education'],trace = df['Occupation'],response = df['Salary'],ylabel='Salary',xlabel='Education')
plt.show()
```




In [5]:
```
model = ols('Salary ~ Education * Occupation',data = df).fit()
anova_table = sm.stats.anova_lm(model)
anova_table
```

Out[5]:

| | df | sum_sq | mean_sq | F | PR(>F) |
|---|---|---|---|---|---|
| Education | 2.0 | 1.026955e+11 | 5.134773e+10 | 72.211958 | 5.466264e-12 |
| Occupation | 3.0 | 5.519946e+09 | 1.839982e+09 | 2.587626 | 7.211580e-02 |
| Education:Occupation | 6.0 | 3.634909e+10 | 6.058182e+09 | 8.519815 | 2.232500e-05 |
| Residual | 29.0 | 2.062102e+10 | 7.110697e+08 | NaN | NaN |

## Problem 2:

In [1]:
```
import numpy as np
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
```

In [4]:
```
df = pd.read_csv('Education+-+Post+12th+Standard.csv')
```

In [5]:
```
df.head()
```

Out[5]:

| | Names | Apps | Accept | Enroll | Top10perc | Top25perc | F.Undergrad | P.Undergrad | Outstate | Room.Board | Books | Personal | PhD | Terminal | S.F.Ratio | perc.. |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | Abilene Christian University | 1660 | 1232 | 721 | 23 | 52 | 2885 | 537 | 7440 | 3300 | 450 | 2200 | 70 | 78 | 18.1 | |
| 1 | Adelphi University | 2186 | 1924 | 512 | 16 | 29 | 2683 | 1227 | 12280 | 6450 | 750 | 1500 | 29 | 30 | 12.2 | |
| 2 | Adrian College | 1428 | 1097 | 336 | 22 | 50 | 1036 | 99 | 11250 | 3750 | 400 | 1165 | 53 | 66 | 12.9 | |

In [6]:
```
df.shape
```

Out[6]: (777, 18)

There are total 777 rows and 18 columns in the dataset

In [7]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 777 entries, 0 to 776
Data columns (total 18 columns):
 #   Column       Non-Null Count  Dtype  
---  ------       --------------  -----  
 0   Names        777 non-null    object 
 1   Apps         777 non-null    int64  
 2   Accept       777 non-null    int64  
 3   Enroll       777 non-null    int64  
 4   Top10perc    777 non-null    int64  
 5   Top25perc    777 non-null    int64  
 6   F.Undergrad  777 non-null    int64  
 7   P.Undergrad  777 non-null    int64  
 8   Outstate     777 non-null    int64  
 9   Room.Board   777 non-null    int64  
 10  Books        777 non-null    int64  
 11  Personal     777 non-null    int64  
 12  PhD          777 non-null    int64  
 13  Terminal     777 non-null    int64  
 14  S.F.Ratio    777 non-null    float64
 15  perc.alumni  777 non-null    int64  
 16  Expend       777 non-null    int64  
 17  Grad.Rate    777 non-null    int64  
dtypes: float64(1), int64(16), object(1)
memory usage: 109.4+ KB
```

In [24]:
```
np.round(df.describe(),2).T
```

Out[24]:

| | count | mean | std | min | 25% | 50% | 75% | max |
|---|---|---|---|---|---|---|---|---|
| Apps | 777.0 | -0.0 | 1.0 | -0.75 | -0.58 | -0.37 | 0.16 | 11.65 |
| Accept | 777.0 | 0.0 | 1.0 | -0.79 | -0.58 | -0.37 | 0.17 | 9.92 |
| Enroll | 777.0 | 0.0 | 1.0 | -0.80 | -0.58 | -0.37 | 0.13 | 6.04 |
| Top10perc | 777.0 | -0.0 | 1.0 | -1.51 | -0.71 | -0.26 | 0.42 | 3.88 |
| Top25perc | 777.0 | -0.0 | 1.0 | -2.36 | -0.75 | -0.09 | 0.67 | 2.23 |
| F.Undergrad | 777.0 | 0.0 | 1.0 | -0.73 | -0.56 | -0.41 | 0.06 | 5.76 |
| P.Undergrad | 777.0 | -0.0 | 1.0 | -0.56 | -0.50 | -0.33 | 0.07 | 13.78 |
| Outstate | 777.0 | 0.0 | 1.0 | -2.01 | -0.78 | -0.11 | 0.62 | 2.80 |
| Room.Board | 777.0 | 0.0 | 1.0 | -2.35 | -0.69 | -0.14 | 0.63 | 3.43 |
| Books | 777.0 | -0.0 | 1.0 | -2.75 | -0.48 | -0.30 | 0.31 | 10.85 |
| Personal | 777.0 | -0.0 | 1.0 | -1.61 | -0.72 | -0.21 | 0.53 | 8.06 |
| PhD | 777.0 | 0.0 | 1.0 | -3.96 | -0.65 | 0.14 | 0.76 | 1.86 |
| Terminal | 777.0 | -0.0 | 1.0 | -3.78 | -0.59 | 0.16 | 0.84 | 1.38 |
| S.F.Ratio | 777.0 | -0.0 | 1.0 | -2.93 | -0.65 | -0.12 | 0.61 | 6.50 |
| perc.alumni | 777.0 | 0.0 | 1.0 | -1.84 | -0.79 | -0.14 | 0.67 | 3.33 |
| Expend | 777.0 | 0.0 | 1.0 | -1.24 | -0.56 | -0.25 | 0.22 | 8.92 |
| Grad.Rate | 777.0 | 0.0 | 1.0 | -3.23 | -0.73 | -0.03 | 0.73 | 3.06 |

In [9]:
```
df.drop('Names',axis = 1,inplace = True)
```

Dropping Names variable as it does not add any significant value to our modelling.

**UNIVARIATE ANALYSIS**

In [10]:
```
cont_cols = list(df.columns)
for col in cont_cols:
    print(col)
    print('Skew :',np.round(df[col].skew(),2))
    plt.figure(figsize=(25,6))
    plt.subplot(1,2,1)
    sns.distplot(df[col],norm_hist=False,kde = False,bins = 50,hist_kws=dict(edgecolor="black", linewidth=1.5))
    plt.vlines(df[col].mean(),ymin = 0, ymax = 40,color = 'red',linewidth = 3)
    plt.vlines(df[col].median(),ymin = 0, ymax = 40,color = 'green',linewidth = 3)
    plt.ylabel('count')
    plt.subplot(1,2,2)
    sns.boxplot(df[col])
    plt.show()
```

In [11]:
```
plt.figure(figsize=(15,7))
mask = np.triu(np.ones_like(df.corr(), dtype=bool))
sns.heatmap(df.corr(),annot = True,mask=mask,cmap="YlGnBu")
```

In [12]:
```
# Variance of unscaled data
np.round(np.var(df,ddof=1),2)
```

Out[12]:
```
Apps           14978459.53
Accept          6007959.70
Enroll           863368.39
Top10perc           311.18
Top25perc           392.23
F.Undergrad    23526579.33
P.Undergrad     2317798.85
Outstate       16184661.63
Room.Board      1202743.03
Books             27259.78
Personal         458425.75
PhD                 266.61
Terminal            216.75
S.F.Ratio            15.67
perc.alumni         153.56
Expend         27266865.64
Grad.Rate           295.07
dtype: float64
```

```
from scipy.stats import zscore
df = pd.DataFrame(zscore(df,ddof=1),columns=cont_cols)
df.head().T
```

In [14]:
```
fig, (ax1, ax2) = plt.subplots(1,2,figsize = (25,6))
mask = np.triu(np.ones_like(df.corr(), dtype=bool))
sns.heatmap(df.corr(),annot = True,mask=mask,ax = ax1,cmap="YlGnBu")
ax1.title.set_text("CORRELATION HEATMAP")
mask = np.triu(np.ones_like(df.cov(), dtype=bool))
sns.heatmap(df.cov(),annot = True,mask=mask,ax = ax2,cmap="YlGnBu")
ax2.title.set_text("COVARIANCE HEATMAP")
plt.show()
```

In [15]:
```
plt.figure(figsize=(18,6))
df.boxplot()
plt.xticks(rotation = 45)
```

In [26]:
```
# Calculation of pc components manually using numpy module
#np.cov calculates the covariance row-wise. So we transpose the dataset to represent the features as rows.

pc_comps = ['PC1','PC2','PC3','PC4','PC5','PC6','PC7','PC8','PC9','PC10','PC11','PC12','PC13','PC14','PC15','PC16','PC17']
cov_matrix = np.cov(df.T)
eigen_values,eigen_vectors_trnspsd = np.linalg.eig(cov_matrix)
pcs = eigen_vectors_trnspsd.T
pc_df_numpy = pd.DataFrame(np.round(pcs,2),index=pc_comps,columns=cont_cols)
pc_df_numpy.head(10).T
```

```
np.round(eigen_vectors_trnspsd,2)
```

In [36]:
```
pc_df_numpy.loc['PC1',:].index
```

Out[36]:
```
Index(['Apps', 'Accept', 'Enroll', 'Top10perc', 'Top25perc', 'F.Undergrad',
       'P.Undergrad', 'Outstate', 'Room.Board', 'Books', 'Personal', 'PhD',
       'Terminal', 'S.F.Ratio', 'perc.alumni', 'Expend', 'Grad.Rate'],
      dtype='object')
```

In [41]:
```
for i,j in zip(pc_df_numpy.loc['PC1',:].index,pc_df_numpy.loc['PC1',:].values):
    print('(',j,')', "*", i,end=' + ')
```

```
( -0.25 ) * Apps + ( -0.21 ) * Accept + ( -0.18 ) * Enroll + ( -0.35 ) * Top10perc + ( -0.34 ) * Top25perc + ( -0.15 ) * F.Unde
rgrad + ( -0.03 ) * P.Undergrad + ( -0.29 ) * Outstate + ( -0.25 ) * Room.Board + ( -0.06 ) * Books + ( 0.04 ) * Personal + ( -
0.32 ) * PhD + ( -0.32 ) * Terminal + ( 0.18 ) * S.F.Ratio + ( -0.21 ) * perc.alumni + ( -0.32 ) * Expend + ( -0.25 ) * Grad.Ra
te +
```

In [21]:
```
tot = np.sum(sorted_eig_vals)
var_exp = [(i/tot) * 100 for i in sorted_eig_vals]
cum_var_exp = np.cumsum(var_exp)
print('Explained variance by each principal component : \n',np.round(var_exp,2))
```

```
Explained variance by each principal component :
 [32.02 26.34  6.9   5.92  5.49  4.98  3.56  3.45  3.12  2.38  1.84  1.3
  0.99  0.85  0.52  0.22  0.14]
```

In [22]:
```
print('Cumulative explained variance : \n',np.round(cum_var_exp,2))
```

```
Cumulative explained variance :
 [ 32.02  58.36  65.26  71.18  76.67  81.66  85.22  88.67  91.79  94.16
  96.    97.3   98.29  99.13  99.65  99.86 100.  ]
```

In [23]:
```
plt.figure(figsize=(10,5))
plt.plot(cum_var_exp,marker = 'o')
plt.xticks(np.arange(0,17),labels=np.arange(1,18))
plt.xlabel('# of principal components')
plt.ylabel('% cumulative variance explained')
```

Great Learning

Happy Learning!